San Francisco Employee Data Prediction

Utkarsha Vidhale

Introduction

- One of the most important aspects of running your business is keeping your employees happy by offering them high-quality employee benefits and compensation.

- So, there must be some solution in which company can know in advance about the compensation structure based on job profile and organization.

- Employers can use this model to imbibe some knowledge regarding the compensation factors and employees can use it to decide which job profiles are receiving maximum benefits

Dataset

Our dataset has 1 file with 835308 instances and 22 columns.

	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Organizati	Organizati	Departme	Departme	Union Coc	Union	Job Famil	Job Famil	Job Code	Job	Employee	Salaries	Overtime	Other Sal	Total Sala	Retiremei	Health an	Other Ber	Total Ben	Total Compensa	
2	7	General C	229259		792	Utd Pub E	0	Untitled	420C	Deputy Cc	8540990	674.28	0	5.76	680.04	130.91	0	53.86	184.77	864.81	
3	1	Public Pro	CRT		792	Utd Pub E	0	Untitled	420C	Deputy Cc	8540990	674.28	0	5.76	680.04	130.91	0	53.86	184.77	864.81	
4	1	Public Pro	CRT		792	Utd Pub E	0	Untitled	420C	Deputy Cc	8540990	674.28	0	5.76	680.04	130.91	0	53.86	184.77	864.81	
5	7	General C	232108		911	POA	Q000	Police Ser	Q004	Police Off	8577148	124709	100499.6	5501.78	230710.4	23271.86	14293.6	3934	55975.56	286686	
6	1	Public Pro	DAT		311	Municipal	8100	Legal & Cc	8177	Attorney (	8603109	155489	0	1500	156989	29239.75	14308.46	11100.6	69326.83	226315.8	
7	7	General C	102644		130	Auto Macl	7300	Journeym	7313	Automoti	8547213	69490.84	34969.05	13344.53	117804.4	16424.93	14308.44	9651.75	48573.42	166377.8	
8	7	General C	DEM		790	SEIU, Loca	8200	Protectior	8238	Public Saf	8544058	57062.72	6033.18	1192	64287.9	11851.05	14308.4	5473.16	33664.43	97952.33	
9	7	General C	102644		253	TWU, Loca	9100	Street Tra	9163	Transit Op	8504938	74231.85	22440.63	3619.49	100292	14778.6	14634.18	7600.99	52233.11	152525.1	

- Predicting Salary
- Predicting Total Compensation

Preprocessing

Preprocessing	Column Names
Negative Values	Salaries, Overtime, Other Salaries, Retirement, Other Benefits
Blanks, Missing Values	Department Code, Union
Removal of unnecessary Columns	Employee Identifier Job family code Union code Organization group code Job code

- Checking negative values

```
[1] 16
> subdata$salaries[subdata$salaries < 0]=mean(subdata$salaries)
> nrow(subdata[subdata$salaries<0,])
[1] 0
> nrow(subdata[subdata$overtime <0,])
[1] 14
> subdata$overtime[subdata$overtime < 0]=mean(subdata$overtime)
> nrow(subdata[subdata$overtime <0,])
[1] 0
> nrow(subdata[subdata$other_salaries <0,])
[1] 17
> subdata$other_salaries[subdata$other_salaries < 0]=mean(subdata$other_salaries)
> nrow(subdata[subdata$other_salaries <0,])
[1] 0
> nrow(subdata[subdata$total_salary <0,])
[1] 11
> subdata$total_salary[subdata$total_salary < 0]=mean(subdata$total_salary)
> nrow(subdata[subdata$total_salary <0,])
[1] 0
> nrow(subdata[subdata$retirement <0,])
[1] 82
> subdata$retirement[subdata$retirement < 0]=mean(subdata$retirement)
> nrow(subdata[subdata$retirement <0,])
[1] 0
> nrow(subdata[subdata$health_and_dental <0,])
[1] 53
> subdata$health_and_dental[subdata$health_and_dental < 0]=mean(subdata$health_and_dental)
> nrow(subdata[subdata$health_and_dental <0,])
[1] 0
> nrow(subdata[subdata$other_benefits <0,])
[1] 146
> subdata$other_benefits[subdata$other_benefits < 0]=mean(subdata$other_benefits)
> nrow(subdata[subdata$other_benefits <0,])
[1] 0
> nrow(subdata[subdata$total_benefits <0,])
[1] 101
> subdata$total_benefits[subdata$total_benefits < 0]=mean(subdata$total_benefits)
```

Issues while loading the dataset

```
Console    Terminal ×    Jobs ×
C:/Users/raina/Downloads/527-Data Analytics/
 $ Employee.Identifier     : int  8540990 8540990 8540990 8577148 8603109 8547213 8544058 8504938 8559329 850
6973 ...
 $ Salaries                : num  674 674 674 124709 155489 ...
 $ Overtime                : num  0e+00 0e+00 0e+00 1e+05 0e+00 ...
 $ Other.Salaries          : num  5.76 5.76 5.76 5501.78 1500 ...
 $ Total.Salary            : num  680 680 680 230710 156989 ...
 $ Retirement              : num  131 131 131 23272 29240 ...
 $ Health.and.Dental       : num  0 0 0 14294 14308 ...
 $ Other.Benefits          : num  53.9 53.9 53.9 3934 11100.6 ...
 $ Total.Benefits          : num  185 185 185 55976 69327 ...
 $ Total.Compensation      : num  865 865 865 286686 226316 ...
> lm(data$Salaries ~ .,data=data)
Error: cannot allocate vector of size 18.2 Gb
> 
```

Solutions :

A) Grouping (Job, Job Family, Union)

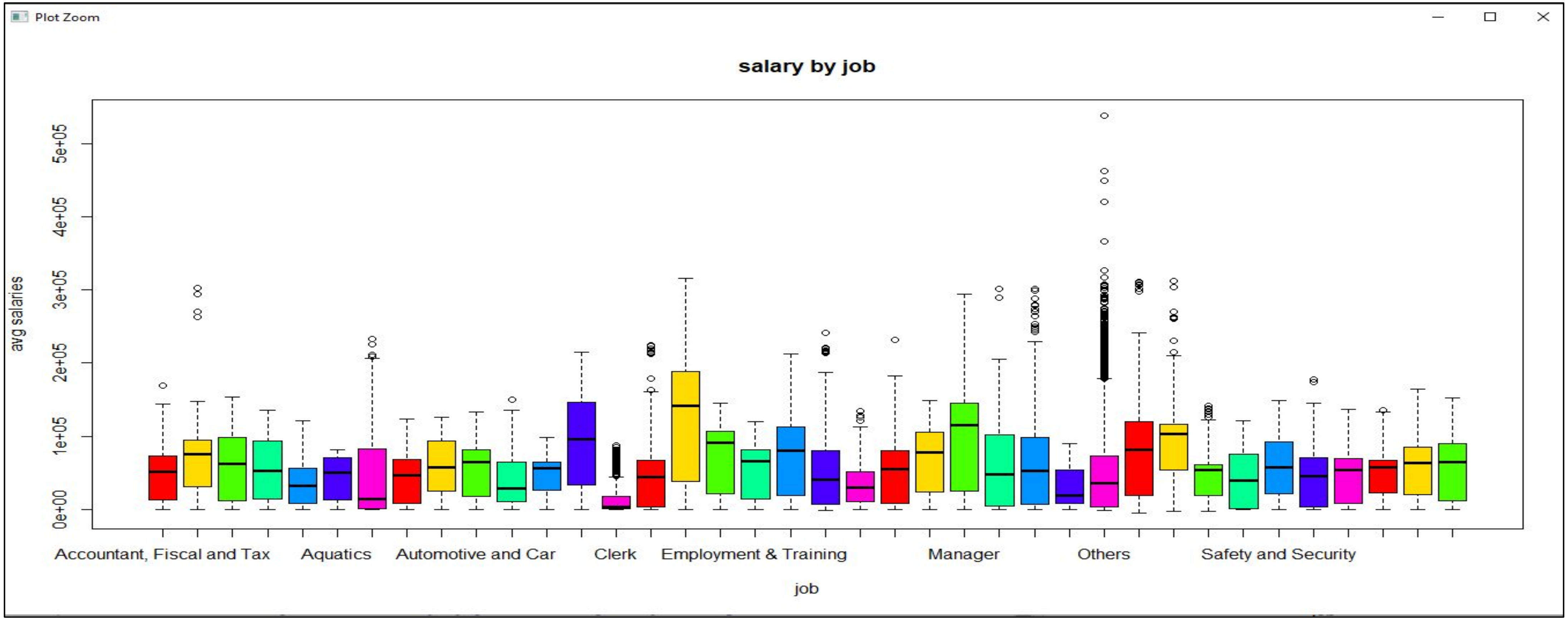
```
> levels(ndata$Job)[levels(ndata$Job)=="Planner 1"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 3"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner V"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 2"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner IV"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 5"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 4"] = "Planners"
```

B) Sampling(150000)

```
> # sampling
> set.seed(5)
> sample_size=150000
> sdata = sample(1:nrow(data),sample_size,replace=F)
> 
```

ANOVA and Hypothesis Testing for Job

- Boxplot for Salaries vs Job



ANOVA and Hypothesis Testing for Job

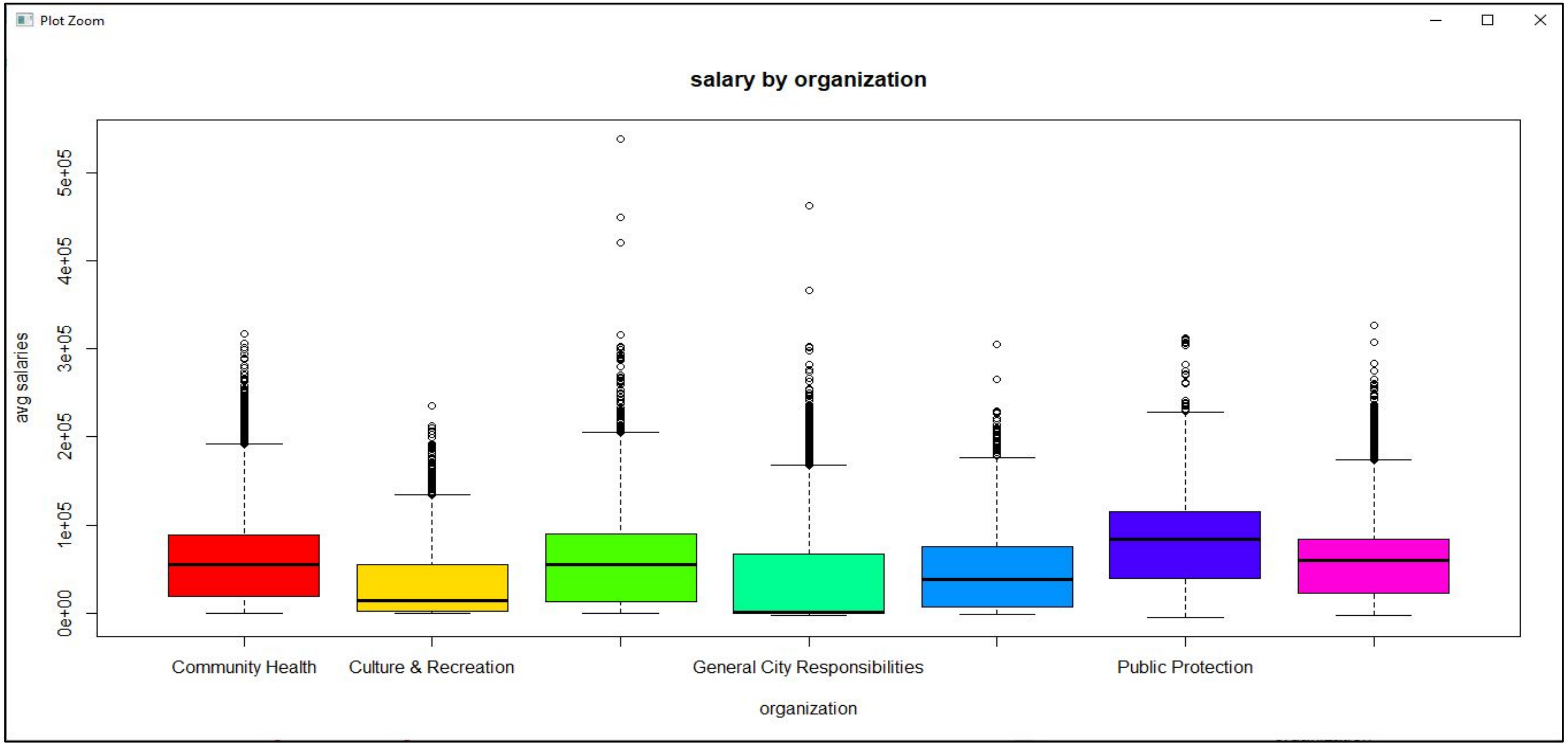
Null hypothesis : All the average salaries for jobs are equal
Alternate hypothesis : Not all the average salaries for jobs are equal

```
> anova(anov)
Analysis of Variance Table

Response: y
              Df     Sum Sq    Mean Sq F value    Pr(>F)
j             37 3.3370e+13 9.0188e+11  434.53 < 2.2e-16 ***
Residuals 149962 3.1125e+14 2.0755e+09
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

At 95% confidence level, p-value is less than 0.05, we can reject null- hypothesis. Hence, the avg salaries are not equal for all jobs.

Mean comparison for Organization_group

- Boxplot for Salaries vs Organization_group



Mean comparison for Organization_group

- We can compare means directly from the boxplots, as variation is almost same in all the cases.

- Public Protection group has the maximum average salaries.

Building Predictive Models:

1. Dummy variables

```
year_type_calendar year_type_fiscal year2014 year2015 year2016 year2017 year2018 year2019
                 0                1        0        0        0        1        0        0
                 1                0        0        0        0        0        0        0
                 1                0        0        0        0        0        1        0
                 0                1        0        1        0        0        0        0
                 0                1        0        1        0        0        0        0
                 1                0        0        0        0        0        0        0
year2028 organization_group_community_health organization_group_culture_recreation
       0                                   0                                     0
       0                                   0                                     0
       0                                   0                                     0
       0                                   0                                     0
       0                                   1                                     0
       0                                   1                                     0
organization_group_general_city_responsibilities
                                               0
                                               0
                                               0
                                               0
                                               0
                                               0
organization_group_human_welfare_neighborhood_development organization_group_public_protection
                                                        0                                    0
                                                        0                                    1
                                                        0                                    0
                                                        0                                    0
                                                        0                                    0
                                                        0                                    0
```

2. Hold Out Evaluation

```
> dim(subdata)
[1] 150000     22
> subdata=subdata[sample(nrow(subdata)),]
> select.data = sample(1:nrow(subdata),0.7*nrow(subdata))
> train.data=subdata[select.data,]
> test.data=subdata[-select.data,]
> dim(train.data)
[1] 105000     22
> dim(test.data)
[1] 45000    22
> 
```

- Predicting Total Compensation of each employee based on other factors.

3. Weak Co relations and Transformation

```
#transformation for overtime and other_salaries
t=compdata$overtime*compdata$overtime
cor(compdata$total_compensation,t, method = "pearson")
t=log(compdata$overtime)
cor(compdata$total_compensation,t, method = "pearson")
t=1/(compdata$overtime)
cor(compdata$total_compensation,t, method = "pearson")
compdata=select(compdata,-c(overtime))

t=compdata$other_salaries*compdata$other_salaries
cor(compdata$total_compensation,t, method = "pearson")
t=log(compdata$other_salaries)
cor(compdata$total_compensation,t, method = "pearson")
t=1/(compdata$other_salaries)
cor(compdata$total_compensation,t, method = "pearson")
compdata=select(compdata,-c(other_salaries))
```

Predicting Total Compensation

- Search Algorithm - **Backward Elimination**, Feature Selection Criteria - AIC

- Model 1

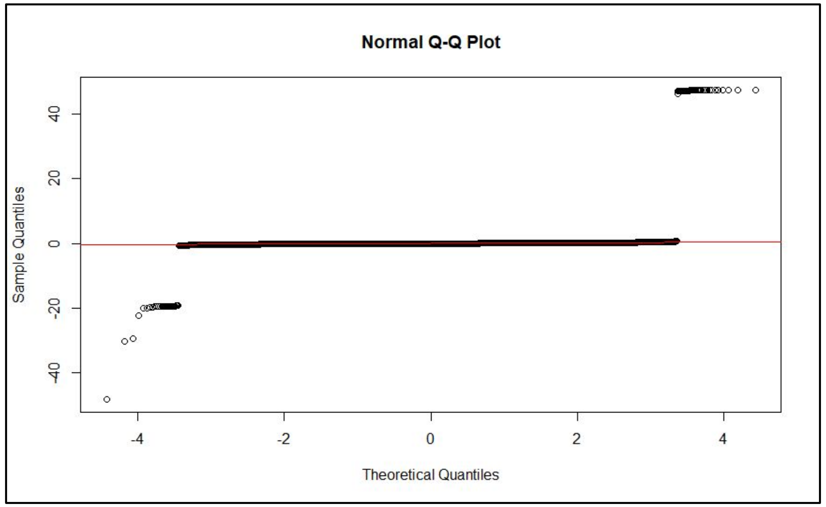
```
#total_compensation
m4=lm(train.data$total_compensation ~ .,data=train.data)
summary(m4)

m3=step(m4, direction = "backward", trace = T)

summary(m3)

# residual analysis

res=rstandard(m3)
```

```
Step:  AIC=-1320830
train.data$total_compensation ~ year_type_calendar + year2015 +
    year2016 + year2017 + year2019 + organization_group_community_health +
    organization_group_culture_recreation + organization_group_human_welfare_neighborhood_
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.001855 on 104952 degrees of freedom
Multiple R-squared:  0.9997,	Adjusted R-squared:  0.9997
F-statistic: 7.592e+06 on 47 and 104952 DF,  p-value: < 2.2e-16
```

Residual Analysis

- Normality test

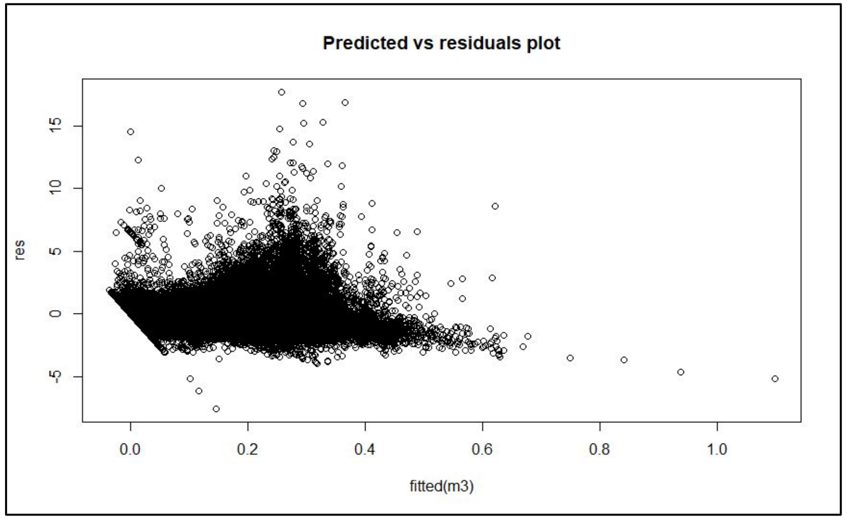


- JarqueBera Test

```
> jarque.bera.test(res)

	Jarque Bera Test

data:  res
X-squared = 1.7737e+10, df = 2, p-value < 2.2e-16
```

Predicting Total Compensation

- Search Algorithm - **Backward Elimination**, Feature Selection Criteria - AIC
- Residual Plot (constant variance)



Checking multi co linearity using VIF

```
> cor(train.data$total_salary,train.data$retirement, method="pearson")
[1] 0.9448298
> cor(train.data$total_salary,train.data$salaries, method="pearson")
[1] 0.9681689
> cor(train.data$total_salary,train.data$organization_group_culture_recreation, method="pearson")
[1] -0.1523838
> cor(train.data$total_salary,train.data$department_code_lib, method="pearson")
[1] -0.04342369
> cor(train.data$total_salary,train.data$department_code_rec, method="pearson")
[1] -0.146247
> cor(train.data$total_salary,train.data$health_and_dental, method="pearson")
[1] 0.5831619
> cor(train.data$total_salary,train.data$other_benefits, method="pearson")
[1] 0.711008
> cor(train.data$total_salary,train.data$total_benefits, method="pearson")
[1] 0.900791
> cor(train.data$total_salary,train.data$department_code_fam, method="pearson")
[1] -0.03540061
> cor(train.data$organization_group_culture_recreation,train.data$total_benefits, method="pearson")
[1] -0.1372857
> cor(train.data$organization_group_culture_recreation,train.data$retirement, method="pearson")
[1] -0.1457864
> cor(train.data$organization_group_culture_recreation,train.data$salaries, method="pearson")
[1] -0.1441378
> cor(train.data$organization_group_culture_recreation,train.data$department_code_lib, method="pearson")
[1] 0.4834578
> cor(train.data$organization_group_culture_recreation,train.data$department_code_rec, method="pearson")
[1] 0.7805858
> cor(train.data$organization_group_culture_recreation,train.data$health_and_dental, method="pearson")
[1] -0.1159762
> cor(train.data$organization_group_culture_recreation,train.data$other_benefits, method="pearson")
[1] -0.09886695
> cor(train.data$organization_group_culture_recreation,train.data$department_code_fam, method="pearson")
[1] 0.2408308
> cor(train.data$salaries,train.data$total_benefits, method="pearson")
[1] 0.9218288
```

```
> train.data=select(train.data,-c(retirement))
> train.data=select(train.data,-c(total_salary))
> train.data=select(train.data,-c(total_benefits))
> train.data=select(train.data,-c(salaries))
>
> test.data=select(test.data,-c(retirement))
> test.data=select(test.data,-c(total_salary))
> test.data=select(test.data,-c(total_benefits))
> test.data=select(test.data,-c(salaries))
```

Removal of columns

Predicting Total Compensation

- Search Algorithm - **Backward Elimination**, Feature Selection Criteria - AIC
  - Model After resolving multi-collinearity

```
> #build model again after removing multicoll
> m5=lm(train.data$total_compensation ~ .,data=train.data)
> summary(m5)

Call:
lm(formula = train.data$total_compensation ~ ., data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.42101 -0.01761 -0.00060  0.01786  0.48592
```

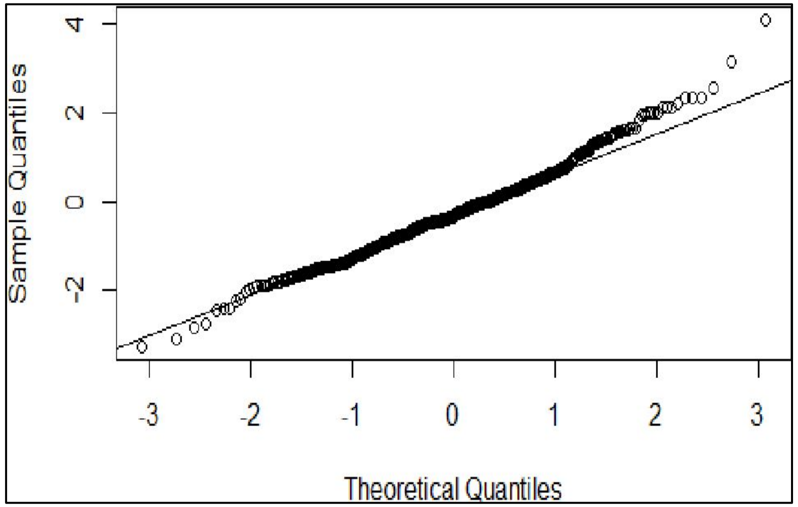
```
Step:  AIC=-657467.7
train.data$total_compensation ~ year_type_calendar + year2015 +
    year2016 + year2017 + year2018 + year2019 + organization_group_community_health +
    organization_group_culture_recreation + organization_group_general_city_responsibi
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04366 on 104835 degrees of freedom
Multiple R-squared:  0.8373,    Adjusted R-squared:  0.837
F-statistic:  3289 on 164 and 104835 DF,  p-value: < 2.2e-16
```

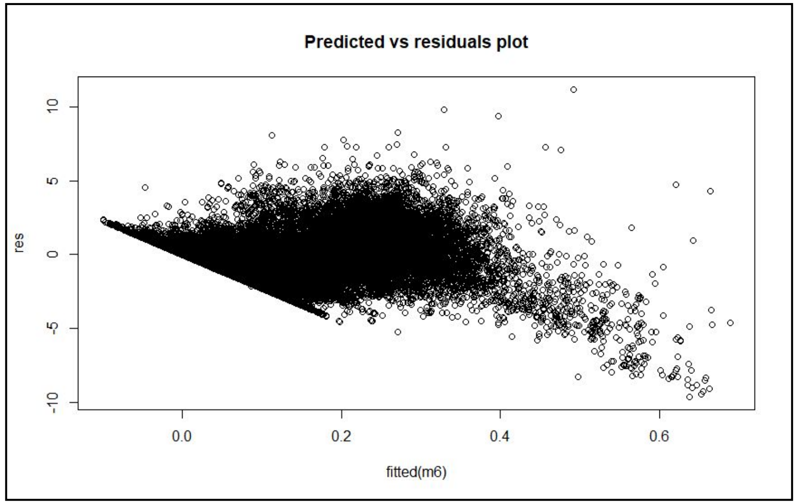
Predicting Total Compensation (Residual Analysis)

- Search Algorithm - **Backward Elimination**, Feature Selection Criteria - AIC

- Normality test



- Residual Plot (constant variance)



- JarqueBera Test

```
> jarque.bera.test(res)

	Jarque Bera Test

data:  res
X-squared = 235949, df = 2, p-value < 2.2e-16

>
```

- RMSE

```
[163] "job_utility_and_janitorial_services"
[164] "job_water_services_and_welfare"
[165] "health_and_dental"
[166] "other_benefits"
[167] "total_compensation"
> y1=predict.glm(m6,test.data)
> y=test.data[,167]
> rmse_2 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_2
           [,1]
[1,] 0.04321609
>
```

Predicting Total Compensation

- Search Algorithm - **Forward Selection**, Feature Selection Criteria - AIC

Similarly we built the final model with forward selection using AIC whose specifications are as below:

- Improved model

```
> base=lm(total_compensation~other_benefits, data=train.data)
> m4=step(base, scope=list(upper=m3, lower=~1),direction="forward",trace=F)
> summary(m4)
```

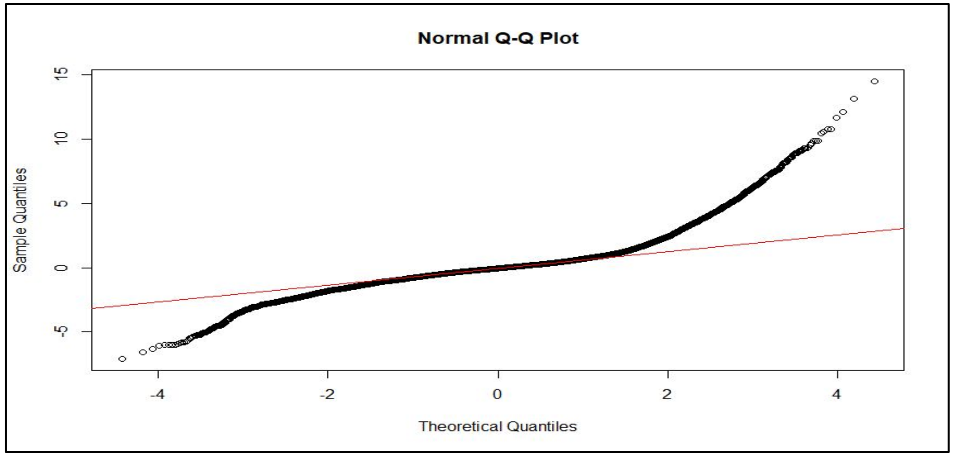
```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02605 on 104835 degrees of freedom
Multiple R-squared:  0.942,	Adjusted R-squared:  0.9419
F-statistic: 1.038e+04 on 164 and 104835 DF,  p-value: < 2.2e-16
```

- Ajd-R2 = 0.942

Predicting Total Compensation (Residual Analysis)

- Search Algorithm - **Forward Selection**, Feature Selection Criteria - AIC

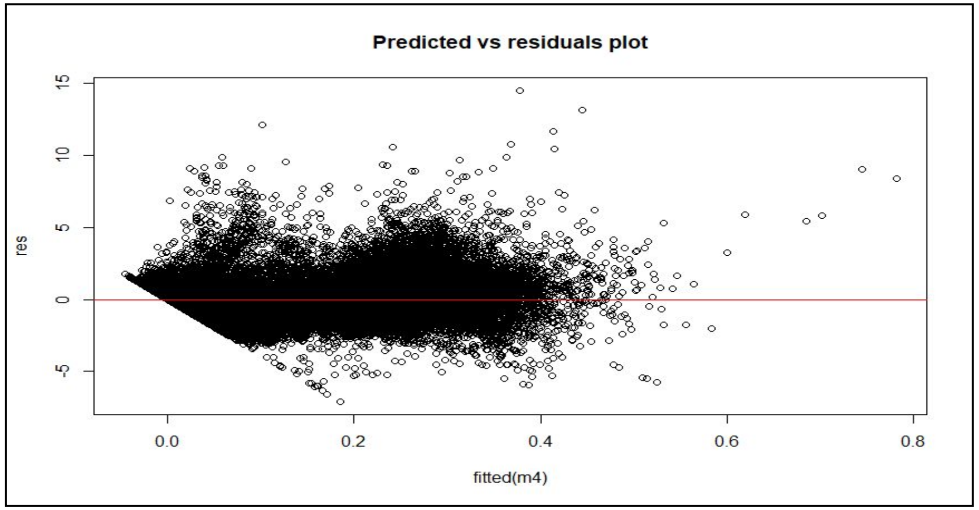
- Normality test



- JarqueBera Test

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 463277, df = 2, p-value < 2.2e-16
```

- Residual plot (Constant Variance)



- RMSE

```
> y1=predict.glm(m4,test.data)
> y=test.data[,168]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_1
           [,1]
[1,] 0.02598934
```

Predicting Total Compensation

- Best Model for Total Compensation
  (Forward Selection And Backward Elimination Comparison)

Measures	Backward Elimination	Forward Selection
ADJ R2	0.837	0.9419
RMSE	0.0431	0.0259

Here, we can see RMSE is better for model with Forward Selection search algorithm. Hence it will be more accurate.

Predicting Salary

- Search Algorithm - **Backward Elimination**, Feature Selection Criteria - AIC

- Like Total Compensation, we built the model for Salary and following are the different metrics we got.

  - Improved model

```
> m2=step(m1, direction = "backward", trace = T)
Start:  AIC=-880753.8
train.data$salaries ~ year_type_calendar + year_type_fiscal +
    year2014 + year2015 + year2016 + year2017 + year2018 + year2019 +
    year2020 + organization_group_community_health + organization_group_c
    organization_group_general_city_responsibilities + organization_group
hood_development +
    organization_group_public_protection + organization_group_public_work
```

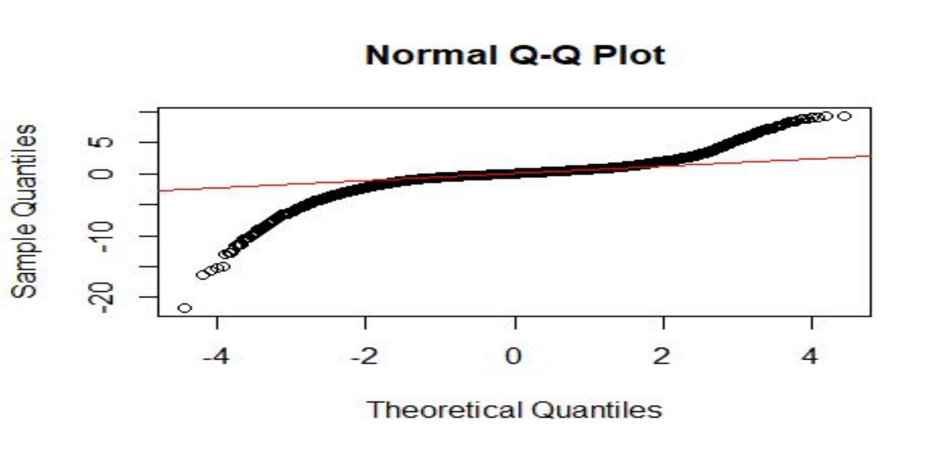
```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.01651 on 104835 degrees of freedom
Multiple R-squared:  0.9659,	Adjusted R-squared:  0.9658
F-statistic: 1.809e+04 on 164 and 104835 DF,  p-value: < 2.2e-16
```

- Ajd-R2 = 0.9958

Predicting Salary (Residual Analysis)

- Search Algorithm – **Backward Elimination**, Feature Selection Criteria - AIC

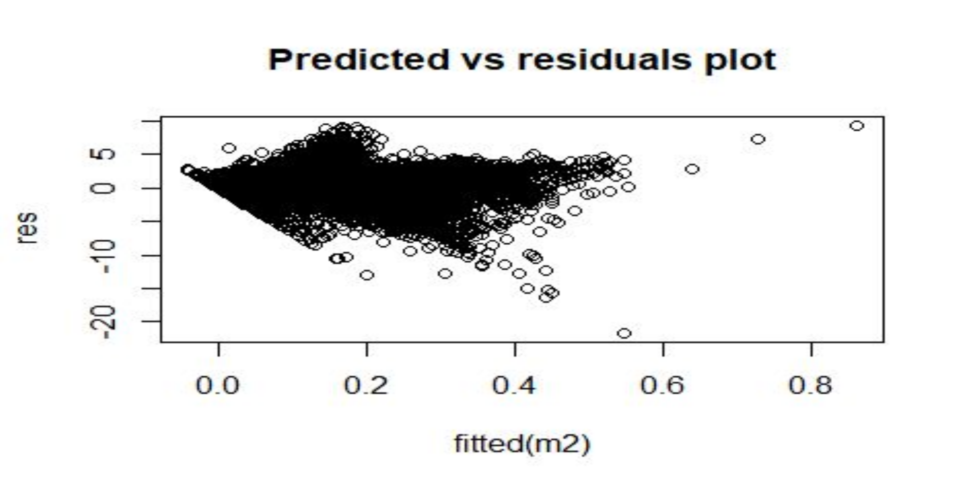
- Normality test



- JarqueBera Test

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1135360, df = 2, p-value < 2.2e-16

>
```

- Residual plot (Constant Variance)



- RMSE

```
> y1=predict.glm(m6,test.data)
> y=test.data[,165]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_1
           [,1]
[1,] 0.01673998
>
```

Predicting Salary

- Search Algorithm - **Forward Selection**, Feature Selection Criteria - AIC

Similarly we built the final model with forward selection whose specifications are as below:

- Improved model

```
#forwad
names(subdata)
base2=lm(salaries~total_compensation, data=train.data)
m4=step(base2, scope=list(upper=m1, lower=~1),direction="forward",trace=F)
summary(m4)
```

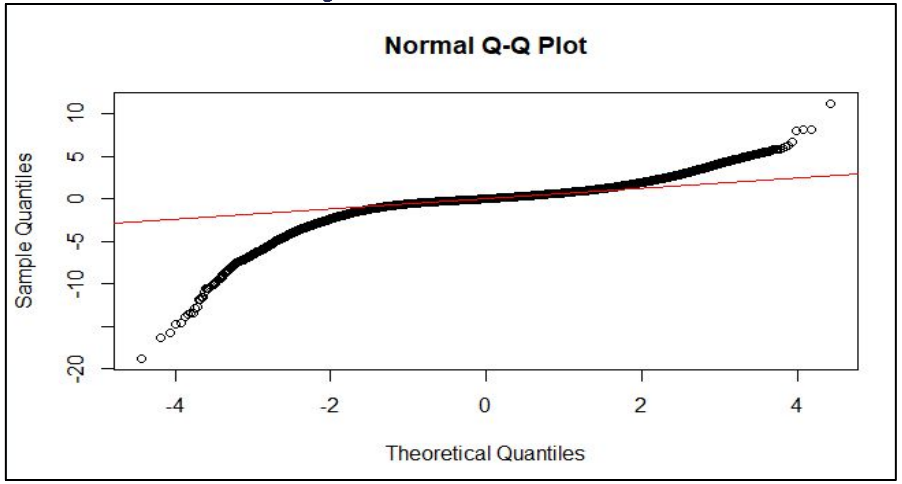
```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01662 on 104881 degrees of freedom
Multiple R-squared:  0.9652,    Adjusted R-squared:  0.9651
F-statistic: 2.462e+04 on 118 and 104881 DF,  p-value: < 2.2e-16

>
```

- Ajd-R2 = 0.9651

Predicting Salary (Residual Analysis)

- Search Algorithm – **Forward Selection**, Feature Selection Criteria - AIC

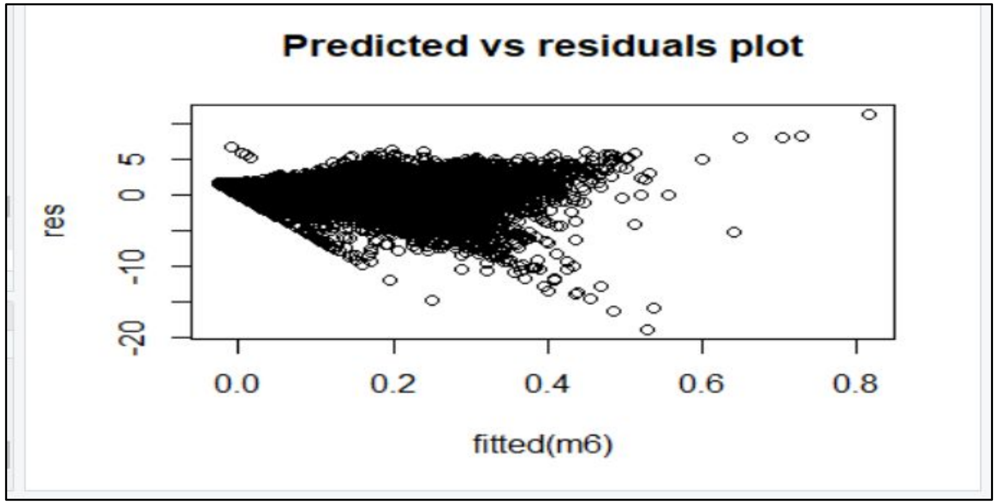
- Normality test



- JarqueBera Test

```
409
410 jarque.bera.test(res)
411
412
413

Console  Terminal  Jobs
C:/Users/utkar/OneDrive/Desktop/da ka/17 nov/
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1522486, df = 2, p-value < 2.2e-16

>
```

- Residual plot (Constant Variance)



- RMSE

```
                                            1.048420
> y=test.data[,164]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_1
           [,1]
[1,] 0.0164827
```

Predicting Salary

- Best Model for Predicting Salary (Forward Selection And Backward Elimination Comparison)

Measures	Backward Elimination	Forward Selection
ADJ R2	0.9658	0.9651
RMSE	0.0167	0.0164

Here, we can see RMSE is slightly better for model with Forward Elimination search algorithm. Hence it will be more accurate.

Limitations And Future Scope

- Grouping of the job profiles in a better way in order to provide best association.
- Individual parameter test for each job profile in ANOVA testing in order to build better prediction model.
- Treatment of influential points; due to large dataset, influence measures wasn't giving proper results for influence points, so we can do it better on proper systems with enhanced specifications.
- Employees can use the predictive model to imply better strategies in terms of better job search which can provide better compensation and salary.
- Similarly, Employers can decide what compensation and salary should be given to the job seeker based on job and other factors in order to optimize their financial status.